

**UNIWERSYTET EKONOMICZNY WE WROCŁAWIU**  
**INFORMATYKA W BIZNESIE**

**INTEGRACJA APLIKACJI I SYSTEMÓW – PROJEKT  
ZALICZENIOWY**

Łukasz Kędzierski (145932)

Łukasz Długosz (145915)

Jakub Proćków (145957)

Prowadzący:

Inż. Paweł Szpak

## **1. Użyte technologie**

Do stworzenia projektu wykorzystany został program Visual Studio 2015, w którym użyto języka C# a obsługa skryptów i eventów został zapewniony za pomocą JavaScript'u, natomiast za obsługę żądań HTTP odpowiedzialny jest ASP.NET – Framework.

## **2. Opis architektury systemu**

W projekcie została użyta aplikacja ASP.NET, w której skład wchodzi:

- Formularze internetowe – WEB FORMS
- Pliki zapewniające logikę aplikacji ( z rozszerzeniem .cs)
- Pliki konfiguracyjne
- Oraz inne pliki typu .html, .css itp.

Cały projekt w głównej mierze opiera się na WEB FORMS'ie jednak wszelkiego rodzaju ustawienia konfiguracyjne odpowiedzialne za działanie aplikacji zawarte są w WEB.CONFIG. Cała logika aplikacji wykonywana jest po stronie serwera za pomocą formularzy internetowych, które to są odpowiedzialne za przesyłanie znaczników ASP.NET oraz całej logiki aplikacji. Dzięki takiemu rozwiązaniu użytkownik otrzymuje spersonalizowane dane w prosty i czytelny sposób.

## **3. Opis dostawców**

Za otrzymywane dane odpowiedzialnych jest dwóch providerów. Pierwszy z nich to MARINE WEATHER API (<https://developer.worldweatheronline.com/api/marine-weather-api.aspx>), który to udostępnia spersonalizowane dane potrzebne do surfowania na podstawie danej długości i szerokości geograficznej wskazanej przez użytkownika. Drugim providerem jest SKI WEATHER API (<https://developer.worldweatheronline.com/api/ski-weather-api.aspx>), który tak samo jak wcześniejsze API udostępnia dane dla narciarzy na podstawie współrzędnych geograficznych. Wyniki wyświetlane w textboxach odnoszą się do przykładowo wybranych zmiennych. Tak naprawdę w odpowiedzi od api otrzymujemy bardzo rozbudowaną prognozę pogodową, rozbita na godziny, posiadającą wiele parametrów. Dla zobrazowania całych prognoz, w plikach projektu zawarte są dwa pliki xml: skii oraz marine pokazujące odpowiedź serwera.

#### **4. Przebieg integracji encji, struktura wynikowa**

Stworzona aplikacja bazuje na bibliotekach GoogleMaps oraz dwóch konkretnych API. Użytkownik klikając na mapę (tworząc event) otrzymuje współrzędne geograficzne. Z tak zagregowanymi danymi aplikacja odwołuje się i tworzy zapytanie HTTP, po czym odpowiednio przygotowane wysyła do SKI oraz MARINE API. Otrzymane odpowiedzi w formacie JSON są serializowane do klas (w naszym przypadku SKI WEATHER i MARINE WEATHER), a następnie metoda zwraca odpowiedź w postaci klasy z wszystkimi informacjami o niej. Konkretnym wynikiem jest dana klasa z odpowiedzią w postaci JSON wyświetlona w textboxie na zaprezentowanej stronie.

#### **5. Napotkane problemy**

Największy problem podczas tworzenia projektu zaistniał w momencie próby wstawienia markera na mapę oraz późniejszym przekazaniu danych geograficznych do konkretnych API. O ile sam zamysł nie był zbyt skomplikowany to problem ten wymagał od naszej grupy dokładnego przeglądnięcia dokumentów udostępnianych od dostawców informacji.

#### **6. Adres do repozytorium**

- <http://freq92-001-site1.ftempurl.com/>
- <https://github.com/Freq92/ASI-projekt>

#### **7. Podział pracy w grupie**

Cały projekt wykonywany był wspólnymi siłami. Zarówno w programowanie jak i w opis zaangażowani byli wszyscy członkowie grupy. Każdy z nas miał różne pomysły oraz konkretne wizje na realizację w/w projektu jednak po wielu dyskusjach udało się wszystko złączyć w całość.

#### **8. Co byśmy zmienili gdybyśmy robili ten projekt jeszcze raz ?**

Gdyby zaistniała możliwość ponownego wykonania projektu na pewno dodalibyśmy możliwość zapisania i zaznaczenia więcej niż jednego markera na mapie dzięki czemu dany użytkownik byłby w stanie porównać otrzymane dane z innym wybranym przez niego miejscem.